US006331973B1

(12) **United States Patent**
Young et al.

(10) **Patent No.:** **US 6,331,973 B1**
(45) **Date of Patent:** **Dec. 18, 2001**

(54) **UNIFYING SLOT ASSIGNMENT PROTOCOL MULTIPLE ACCESS SYSTEM**

(75) Inventors: **C. David Young**, Plano; **Roy W. Krueger**, Bedford; **Michael S. Becker**, Dallas; **Henry O. Harwell**, Garland; **James A. Stevens**, Allen, all of TX (US)

(73) Assignee: **Rockwell Collins, Inc.**, Cedar Rapids, IA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/303,528**

(22) Filed: **Apr. 30, 1999**

(51) **Int. Cl.**[7] **H04B 7/212**

(52) **U.S. Cl.** **370/337**; 370/350; 370/436; 370/442

(58) **Field of Search** 370/321, 310, 370/338, 336, 329, 335, 330, 228, 227, 337, 347, 343, 441, 280, 468, 277, 342

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,696,903 * 12/1997 Mahany 709/228
5,719,868 2/1998 Young 370/436
5,742,593 * 4/1998 Sharony 370/330

OTHER PUBLICATIONS

"Bridging the Gap Interoperability, Survivability, Security" Lawrence Pond and Victor O. K. Li, 1989 IEEE Military Communications Conference, Conference Record, vol. 1 of 3 IEEE Transaction On Information Theory, vol. IT-30, No. 4, Jul. 19984.

"Clique activation Multiple Access (CAMA): A Distributed Heuristic for Building Wireless Datagram Networks" C. D. Young and J. Stevens, IEEE Military Communications Conference, 1998, vol. 1.

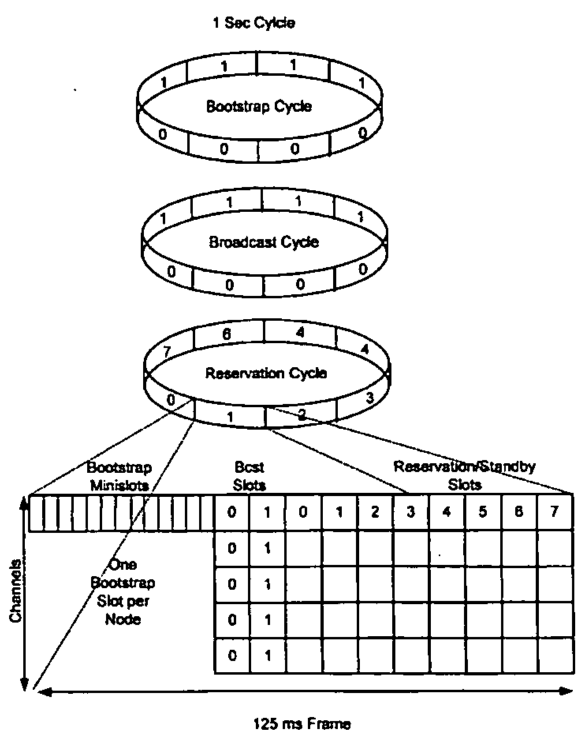
* cited by examiner

Primary Examiner—Douglas Olms
Assistant Examiner—Ricardo M. Pizarro
(74) *Attorney, Agent, or Firm*—Nathan O. Jensen; Kyle Eppele; James P. O'Shaughnessy

(57) **ABSTRACT**

A method and system for managing communication resources between nodes in a network, and more particularly to a dynamic distributed multi-channel time division multiple access (TDMA) slot assignment method is presented. The method and apparatus include a set of higher level heuristics that enable the wireless channel access scheme to address predetermined characteristics of the wireless channel access system. These predetermined heuristics include using bootstrap slots, adaptive broadcast cycles, channelized neighborhoods, standby slots, speculation slots, neighbor segregation, hard circuits, and soft circuits.

39 Claims, 20 Drawing Sheets



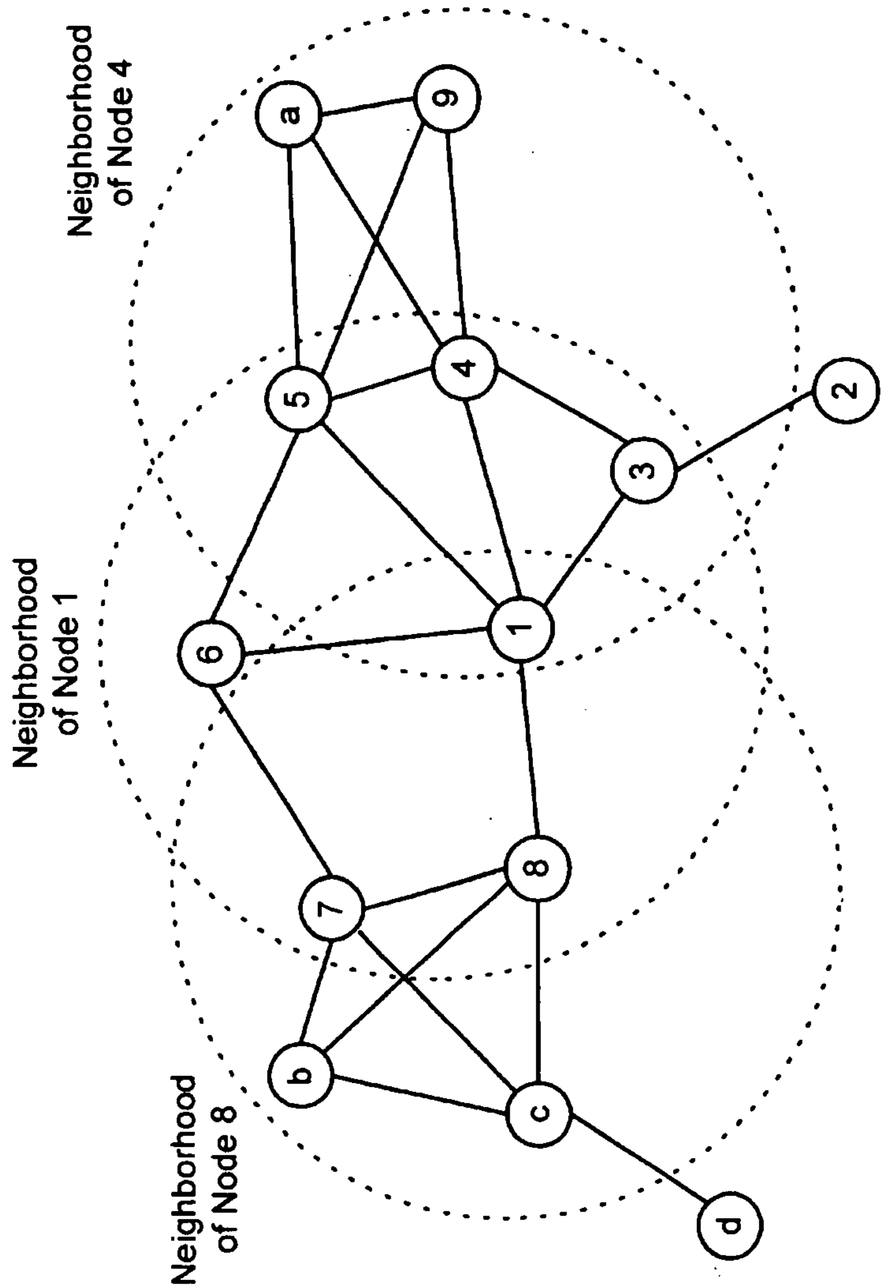
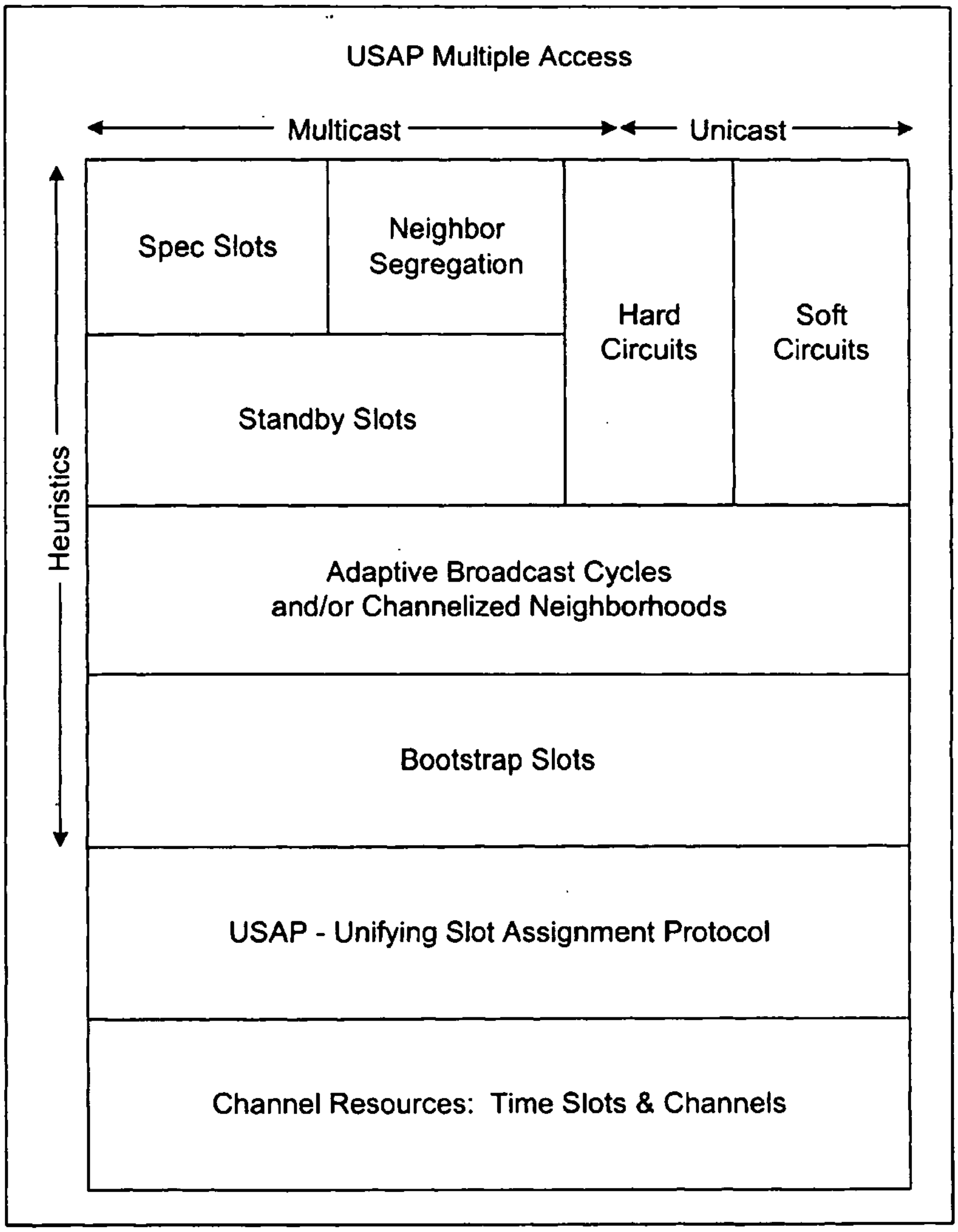
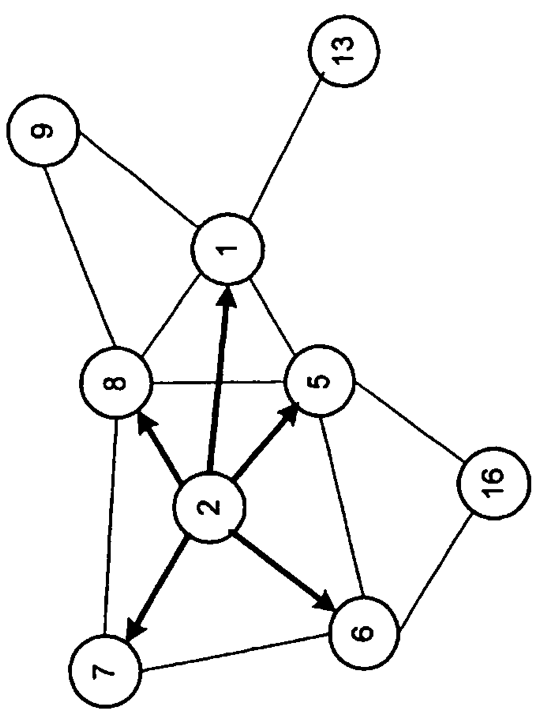

Neighborhood of Node 4
Neighborhood of Node 1
Neighborhood of Node 8
a
9
5
4
3
2
6
1
7
8
b
c
d

FIG. 1



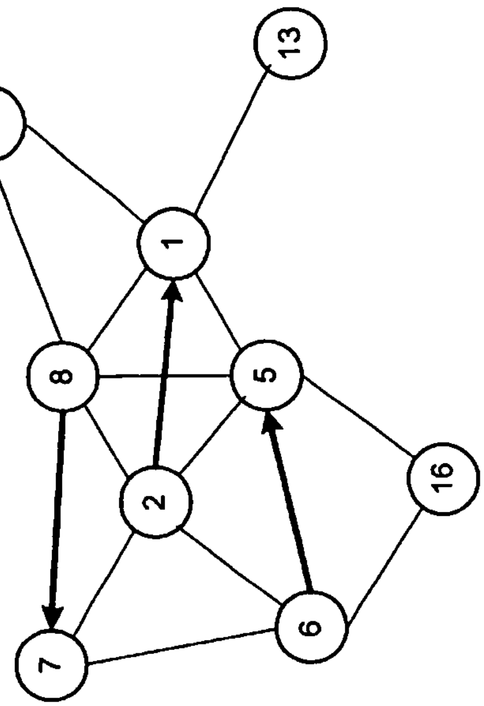
USAP Multiple Access
Multicast
Unicast
Heuristics
Spec Slots
Neighbor Segregation
Hard Circuits
Soft Circuits
Standby Slots
Adaptive Broadcast Cycles and/or Channelized Neighborhoods
Bootstrap Slots
USAP - Unifying Slot Assignment Protocol
Channel Resources: Time Slots & Channels

FIG. 2



Node Activation

FIG. 3



Link Activation

FIG. 4

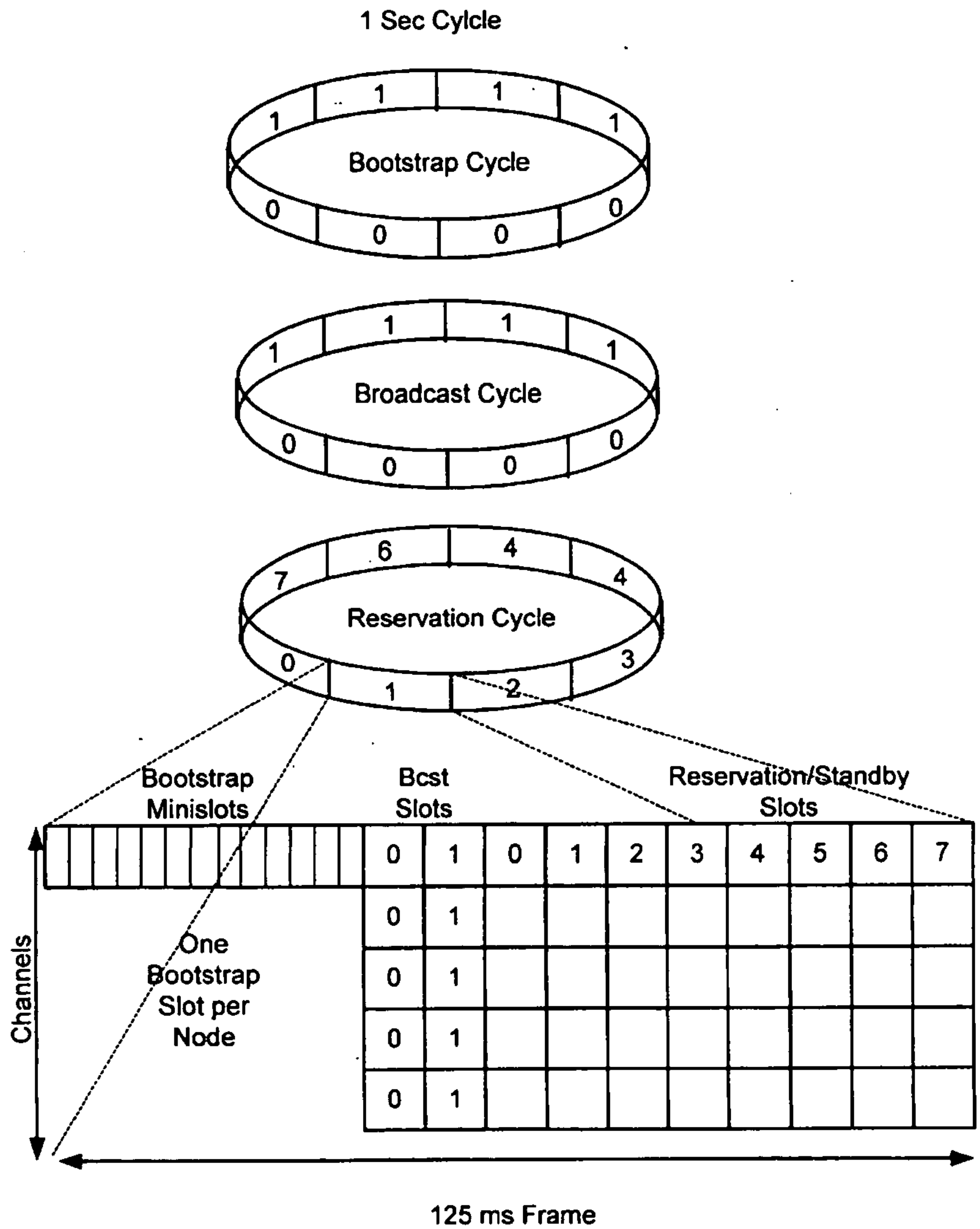

1 Sec Cylcle
Bootstrap Cycle
1
1
1
1
0
0
0
0
Broadcast Cycle
1
1
1
1
0
0
0
0
Reservation Cycle
7
6
4
4
0
1
2
3
Bootstrap Minislots
Bcst Slots
Reservation/Standby Slots
0
1
0
1
2
3
4
5
6
7
Channels
One Bootstrap Slot per Node
125 ms Frame

FIG. 5

USAP-MA Frame Structure and Slot Schedule

← 1 Sec →

	Boostrap	Broadcast		Reservation (Standby Broadcast)							
F0		B0	B1	R0 **(B0)**	R1 (B1)	R2 (B2)	R3 (B3)	R4 (B4)	R5 (B5)	R6 (B6)	R7 (B7)
F1		B2	B3	R0 (B1)	R1 (B2)	R2 (B3)	R3 (B4)	R4 (B5)	R5 (B6)	R6 (B7)	R7 **(B0)**
F2		B4	B5	R0 (B2)	R1 (B3)	R2 (B4)	R3 (B5)	R4 (B6)	R5 (B7)	R6 **(B0)**	R7 (B1)
F3		B6	B7	R0 (B3)	R1 (B4)	R2 (B5)	R3 (B6)	R4 (B7)	R5 **(B0)**	R6 (B1)	R7 (B2)
F4		B0	B1	R0 (B4)	R1 (B5)	R2 (B6)	R3 (B7)	R4 **(B0)**	R5 (B1)	R6 (B2)	R7 (B3)
F5		B2	B3	R0 (B5)	R1 (B6)	R2 (B7)	R3 **(B0)**	R4 (B1)	R5 (B2)	R6 (B3)	R7 (B4)
F6		B4	B5	R0 (B6)	R1 (B7)	R2 **(B0)**	R3 (B1)	R4 (B2)	R5 (B3)	R6 (B4)	R7 (B5)
F7		B6	B7	R0 (B7)	R1 **(B0)**	R2 (B1)	R3 (B2)	R4 (B3)	R5 (B4)	R6 (B5)	R7 (B6)

FIG. 6

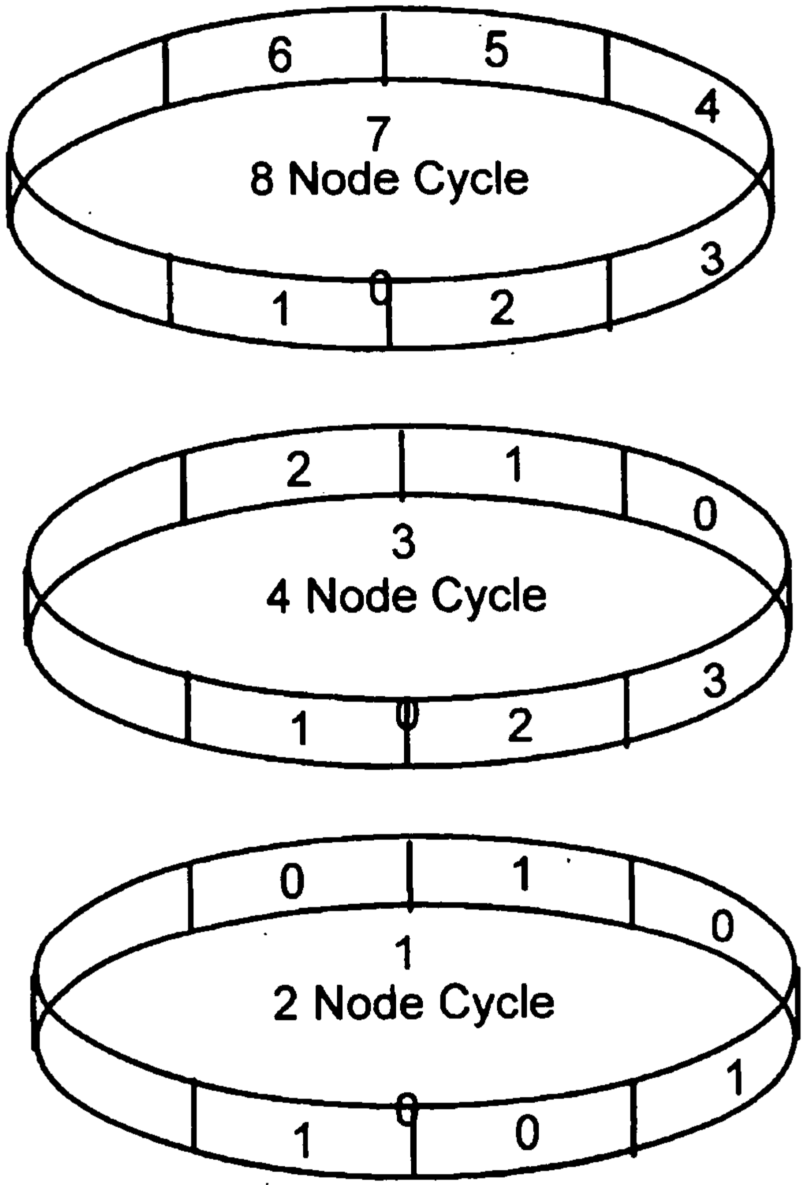

6
5
4
7
8 Node Cycle
3
1
0
2
2
1
0
3
4 Node Cycle
3
1
0
2
0
1
0
1
2 Node Cycle
1
1
0
0

FIG. 7

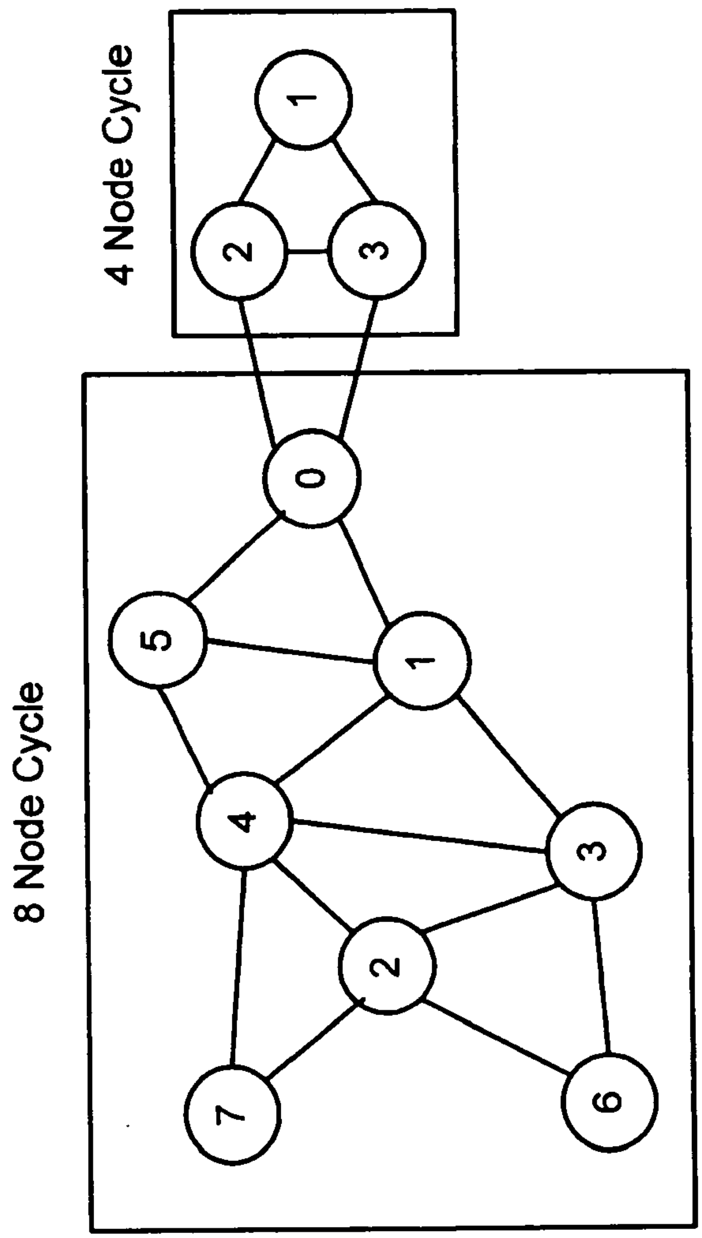


FIG. 8

Broadcast B0	Broadcast B1	Standby Broadcast SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
0	1	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1	0	1
0	1	1	0	1	0	1	0	1	0

2 Node

Broadcast B0	Broadcast B1	Standby Broadcast SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
0	1	0	1	2	3	0	1	2	3
2	3	1	2	3	0	1	2	3	0
0	1	2	3	0	1	2	3	0	1
2	3	3	0	1	2	3	0	1	2
0	1	0	1	2	3	0	1	2	3
2	3	1	2	3	0	1	2	3	0
0	1	2	3	0	1	2	3	0	1
2	3	3	0	1	2	3	0	1	2

4 Node

Fig 9-1

Broadcast		Standby Broadcast							
B0	B1	SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
0	1	0	1	2	3	4	5	6	7
2	3	1	2	3	4	5	6	7	0
4	5	2	3	4	5	6	7	0	1
6	7	3	4	5	6	7	0	1	2
0	1	4	5	6	7	0	1	2	3
2	3	5	6	7	0	1	2	3	4
4	5	6	7	0	1	2	3	4	5
6	7	7	0	1	2	3	4	5	6

8 Node

Broadcast		Standby Broadcast							
B0	B1	SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
0	1	0	1	2	3	4	5	6	7
2	3	8	9	10	11	12	13	14	15
4	5	2	3	4	5	6	7	0	1
6	7	10	11	12	13	14	15	8	9
8	9	4	5	6	7	0	1	2	3
10	11	12	13	14	15	8	9	10	11
12	13	6	7	0	1	2	3	4	5
14	15	14	15	8	9	10	11	12	13

16 Node

Fig 9-2

Broadcast		Standby Broadcast							
B0	B1	SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
0	1	0	1	2	3	4	5	6	7
2	3	8	9	10	11	12	13	14	15
4	5	16	17	18	19	20	21	22	23
6	7	24	25	26	27	28	29	30	31
8	9	4	5	6	7	0	1	2	3
10	11	12	13	14	15	8	9	10	11
12	13	20	21	22	23	16	17	18	19
14	15	28	29	30	31	24	25	26	27

32 Node
1st sec

Broadcast		Standby Broadcast							
B0	B1	SB0	SB1	SB2	SB3	SB4	SB5	SB6	SB7
17	18	0	1	2	3	4	5	6	7
19	20	8	9	10	11	12	13	14	15
21	22	16	17	18	19	20	21	22	23
23	24	24	25	26	27	28	29	30	31
25	26	4	5	6	7	0	1	2	3
27	28	12	13	14	15	8	9	10	11
29	30	20	21	22	23	16	17	18	19
31	32	28	29	30	31	24	25	26	27

32 Node
2nd sec

Fig 9-3

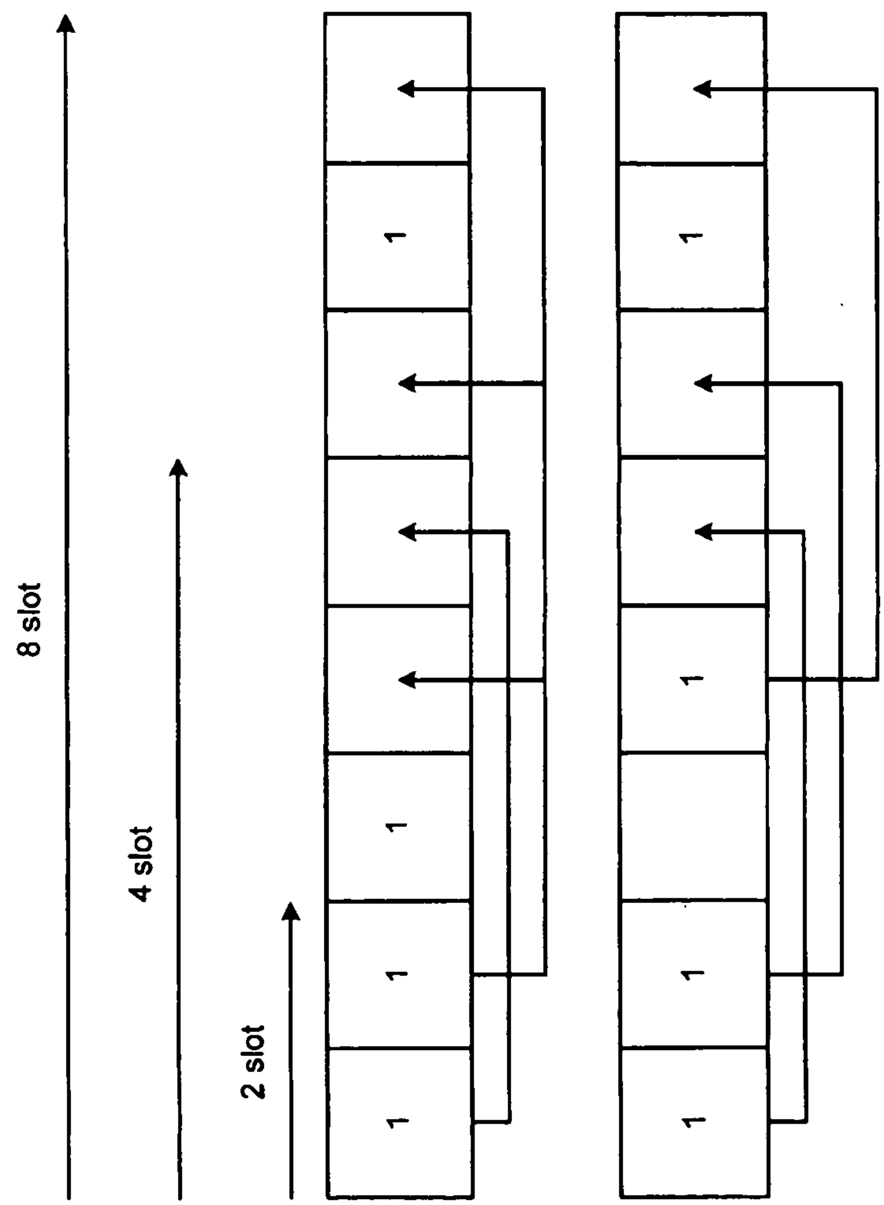


Fig. 10

105

8 Node

105

2 Sec.

		RRG Chan: 0 0 0 0	Bcst		Standby Bcst							
T0	F0	0 1 2 3	0	1	0	1	2	3	4	5	6	7
	F1	1 2 3 0	2	3	1	2	3	4	5	6	7	0
	F2	2 3 0 1	4	5	2	3	4	5	6	7	0	1
	F3	3 0 1 2	6	7	3	4	5	6	7	0	1	2
T0	F0	1 2 3 0	0	1	4	5	6	7	0	1	2	3
	F1	2 3 0 1	2	3	5	6	7	0	1	2	3	4
	F2	3 0 1 2	4	5	6	7	0	1	2	3	4	5
	F3	0 1 2 3	6	7	7	0	1	2	3	4	5	6
T0	F0	2 3 0 1	0	1	0	1	2	3	4	5	6	7
	F1	3 0 1 2	2	3	1	2	3	4	5	6	7	0
	F2	0 1 2 3	4	5	2	3	4	5	6	7	0	1
	F3	1 2 3 0	6	7	3	4	5	6	7	0	1	2
T0	F0	3 0 1 2	0	1	4	5	6	7	0	1	2	3
	F1	0 1 2 3	2	3	5	6	7	0	1	2	3	4
	F2	1 2 3 0	4	5	6	7	0	1	2	3	4	5
	F3	2 3 0 1	6	7	7	0	1	2	3	4	5	6

RRG#: 0 1 2 4

Fig. 11-1

16 Node

110

		RRG Chan: 0 1 0 1	Bcst				Standby Bcst					
T0	F0	0 1 2 3	0	1	0	1	2	3	4	5	6	7
	F1	1 2 3 0	2	3	1	2	3	4	5	6	7	0
	F2	2 3 0 1	4	5	2	3	4	5	6	7	0	1
	F3	3 0 1 2	6	7	3	4	5	6	7	0	1	2
T1	F0	1 2 3 0	0	1	4	5	6	7	0	1	2	3
	F1	2 3 0 1	2	3	5	6	7	0	1	2	3	4
	F2	3 0 1 2	4	5	6	7	0	1	2	3	4	5
	F3	0 1 2 3	6	7	7	0	1	2	3	4	5	6
T2	F0	2 3 0 1	0	1	0	1	2	3	4	5	6	7
	F1	3 0 1 2	2	3	1	2	3	4	5	6	7	0
	F2	0 1 2 3	4	5	2	3	4	5	6	7	0	1
	F3	1 2 3 0	6	7	3	4	5	6	7	0	1	2
T3	F0	3 0 1 2	0	1	4	5	6	7	0	1	2	3
	F1	0 1 2 3	2	3	5	6	7	0	1	2	3	4
	F2	1 2 3 0	4	5	6	7	0	1	2	3	4	5
	F3	2 3 0 1	6	7	7	0	1	2	3	4	5	6

2 Sec.

RRG#: 0 1 2 4

Fig. 11-2

24 Node

115

RRG Chan: 0 1 2 0 — Bcst — Standby Bcst

2 Sec.

		RRG Chan: 0 1 2 0	Bcst				Standby Bcst					
T0	F0	0 1 2 3	0	1	0	1	2	3	4	5	6	7
	F1	1 2 3 0	2	3	1	2	3	4	5	6	7	0
	F2	2 3 0 1	4	5	2	3	4	5	6	7	0	1
	F3	3 0 1 2	6	7	3	4	5	6	7	0	1	2
T1	F0	1 2 3 0	0	1	4	5	6	7	0	1	2	3
	F1	2 3 0 1	2	3	5	6	7	0	1	2	3	4
	F2	3 0 1 2	4	5	6	7	0	1	2	3	4	5
	F3	0 1 2 3	6	7	7	0	1	2	3	4	5	6
T0	F0	2 3 0 1	0	1	0	1	2	3	4	5	6	7
	F1	3 0 1 2	2	3	1	2	3	4	5	6	7	0
	F2	0 1 2 3	4	5	2	3	4	5	6	7	0	1
	F3	1 2 3 0	6	7	3	4	5	6	7	0	1	2
T1	F0	3 0 1 2	0	1	4	5	6	7	0	1	2	3
	F1	0 1 2 3	2	3	5	6	7	0	1	2	3	4
	F2	1 2 3 0	4	5	6	7	0	1	2	3	4	5
	F3	2 3 0 1	6	7	7	0	1	2	3	4	5	6

RRG#: 0 1 2 4

Fig. 11-3

32 Node

100

		RRG Chan: 0 1 2 3	Bcst				Standby Bcst					
T0	F0	0 1 2 3	0	1	0	1	2	3	4	5	6	7
	F1	1 2 3 0	2	3	1	2	3	4	5	6	7	0
	F2	2 3 0 1	4	5	2	3	4	5	6	7	0	1
	F3	3 0 1 2	6	7	3	4	5	6	7	0	1	2
T1	F0	1 2 3 0	0	1	4	5	6	7	0	1	2	3
	F1	2 3 0 1	2	3	5	6	7	0	1	2	3	4
	F2	3 0 1 2	4	5	6	7	0	1	2	3	4	5
	F3	0 1 2 3	6	7	7	0	1	2	3	4	5	6
T0	F0	2 3 0 1	0	1	0	1	2	3	4	5	6	7
	F1	3 0 1 2	2	3	1	2	3	4	5	6	7	0
	F2	0 1 2 3	4	5	2	3	4	5	6	7	0	1
	F3	1 2 3 0	6	7	3	4	5	6	7	0	1	2
T1	F0	3 0 1 2	0	1	4	5	6	7	0	1	2	3
	F1	0 1 2 3	2	3	5	6	7	0	1	2	3	4
	F2	1 2 3 0	4	5	6	7	0	1	2	3	4	5
	F3	2 3 0 1	6	7	7	0	1	2	3	4	5	6

2 Sec.

RRG#: 0 1 2 4

Fig. 11-4

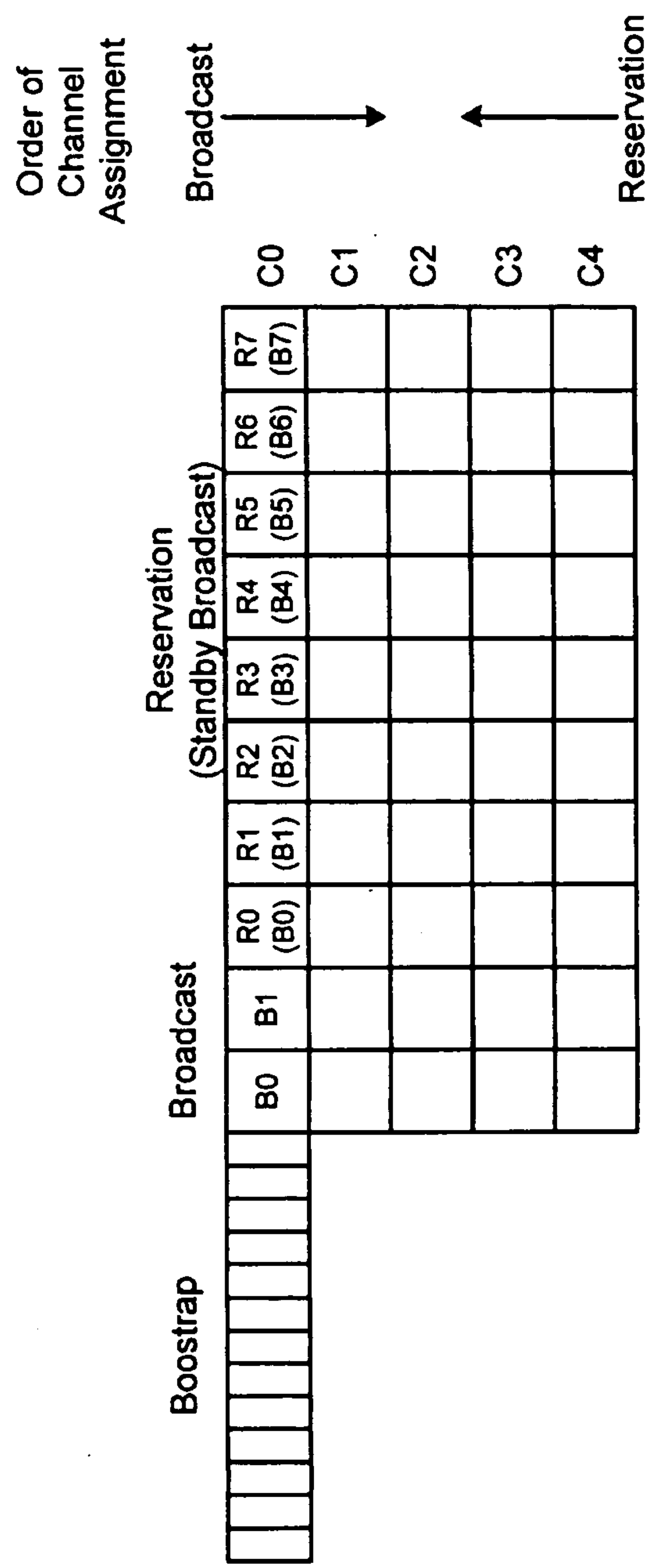

Order of Channel Assignment
Broadcast
Reservation
C0
C1
C2
C3
C4
Reservation (Standby Broadcast)
R7 (B7)
R6 (B6)
R5 (B5)
R4 (B4)
R3 (B3)
R2 (B2)
R1 (B1)
R0 (B0)
Broadcast
B1
B0
Boostrap

FIG. 12

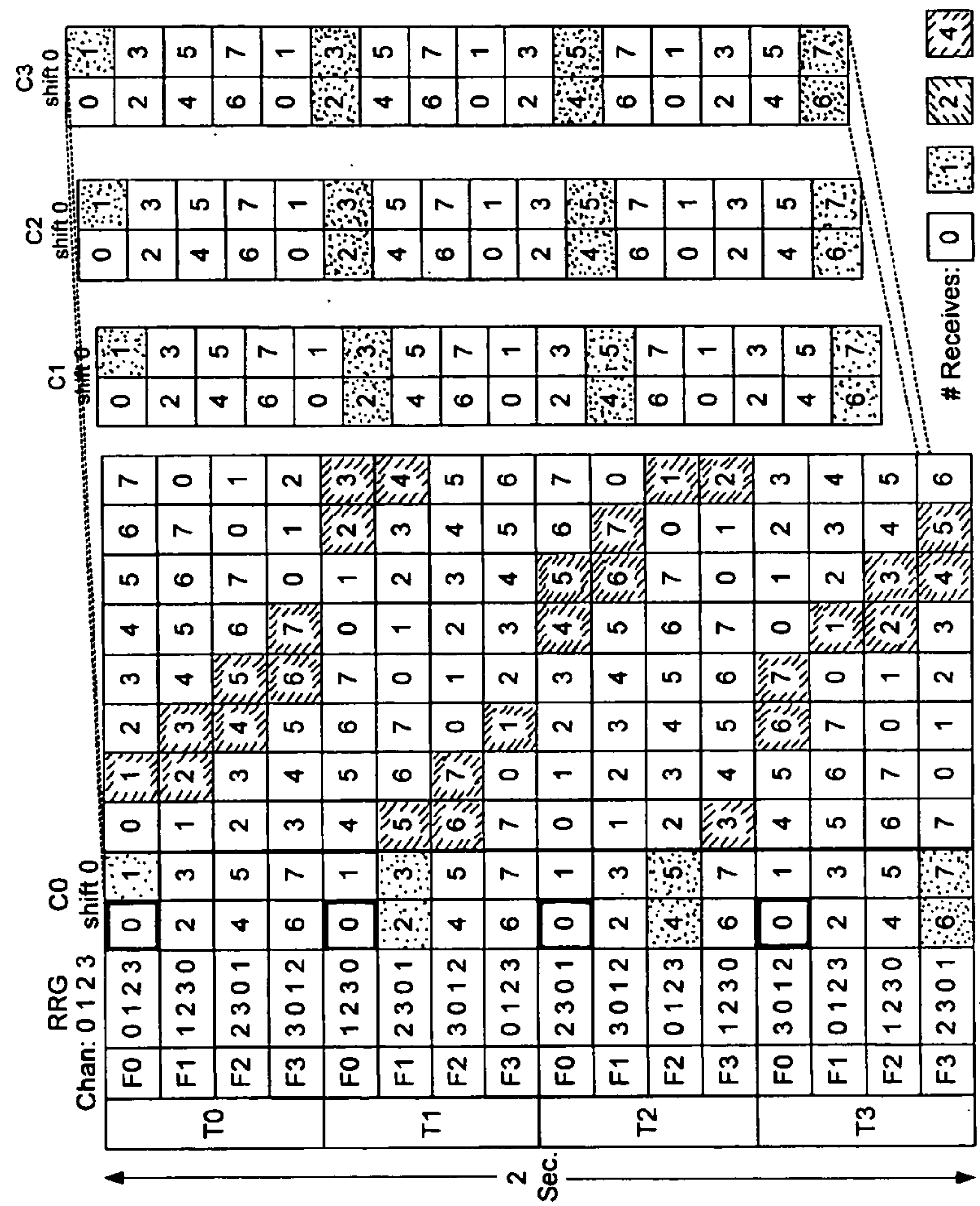

C0 shift 0
C1 shift 0
C2 shift 0
C3 shift 0
Chan:
RRG 0 1 2 3
T0
T1
T2
T3
2 Sec.
Receives:

Fig. 13-1

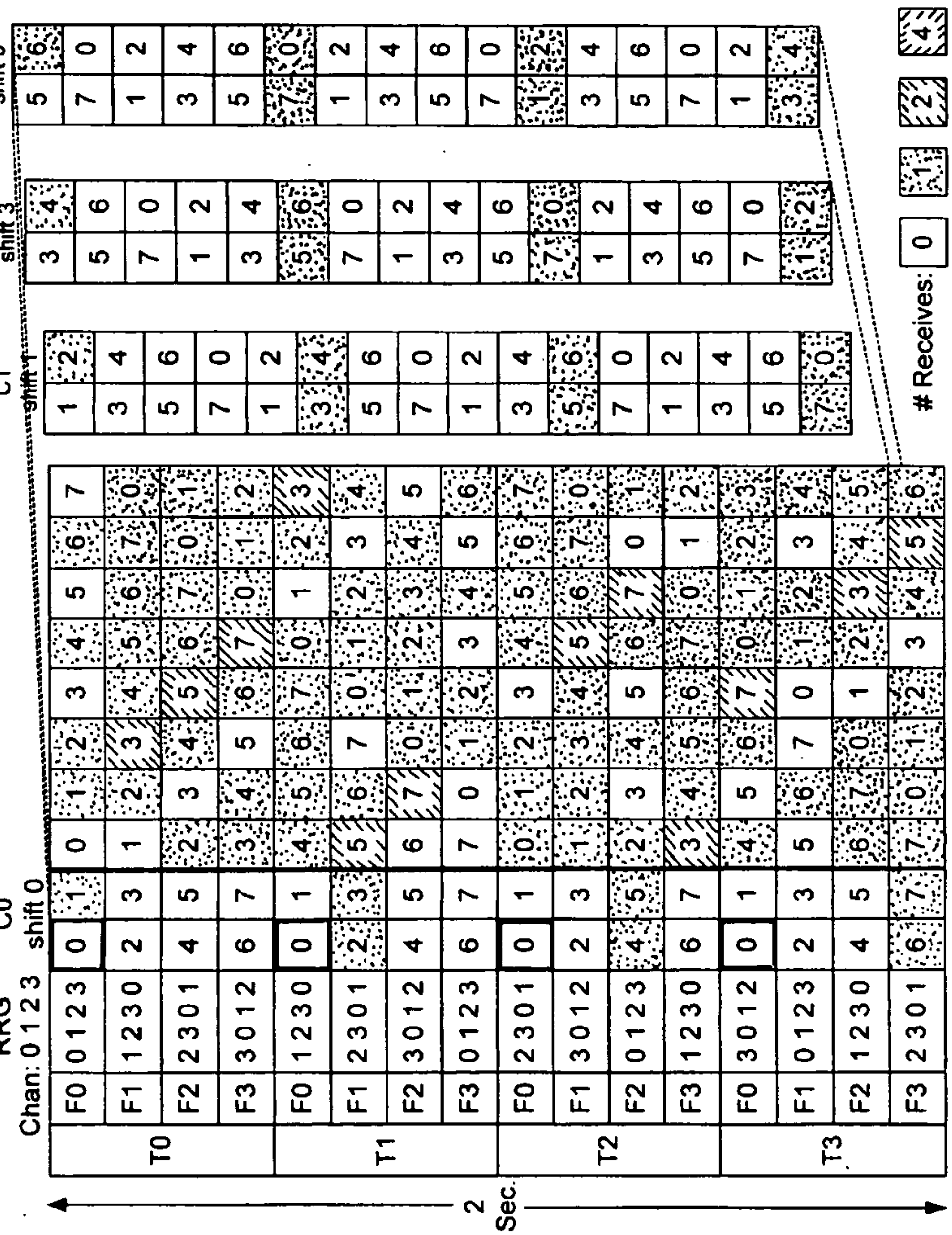

RRG
Chan: 0 1 2 3
C0 shift 0
C1 shift 1
C2 shift 3
C3 shift 5
T0
T1
T2
T3
2 Sec.
Receives:

Fig. 13-2

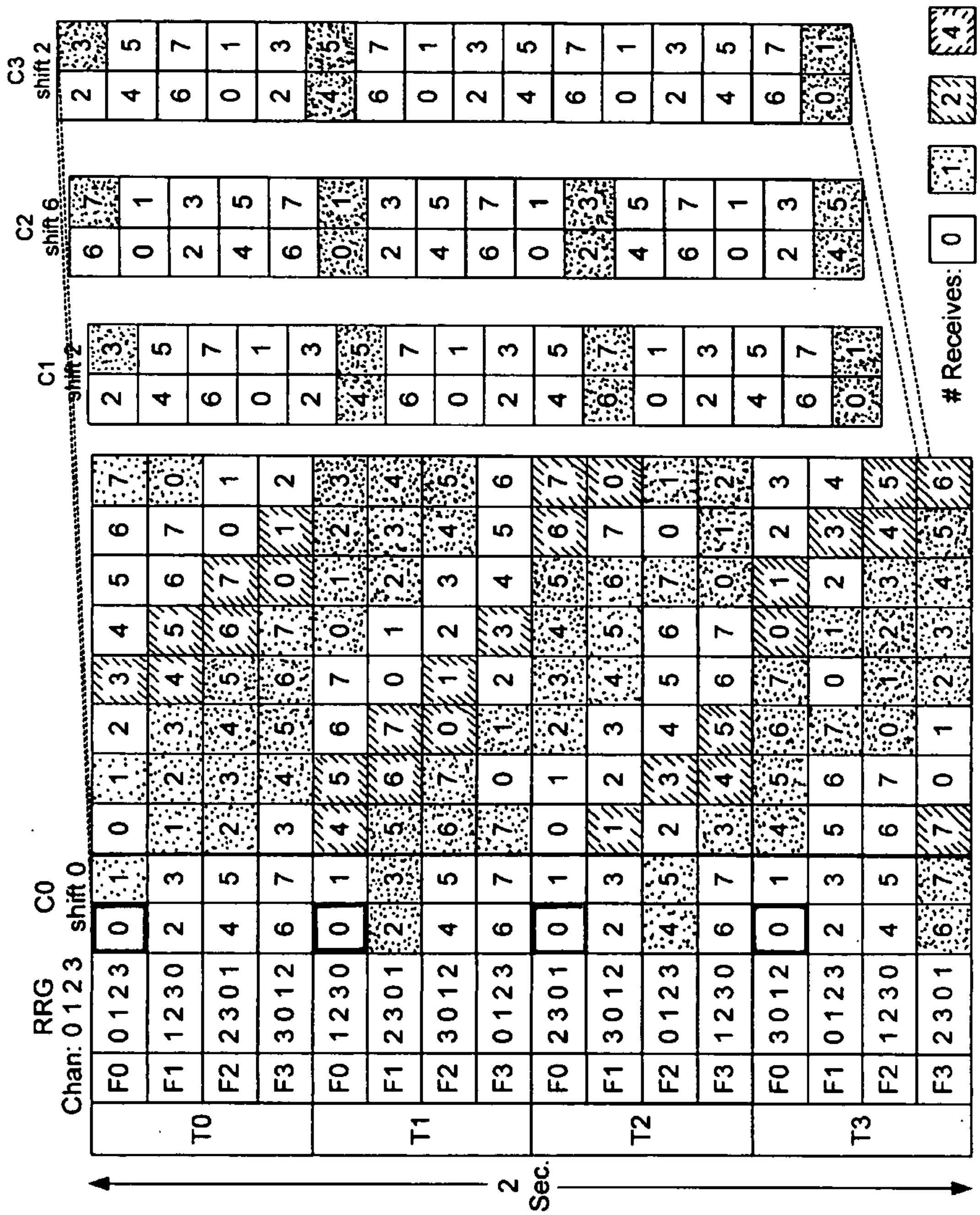

RRG
Chan: 0 1 2 3
C0
shift 0
C1
shift 2
C2
shift 6
C3
shift 2
T0
T1
T2
T3
2 Sec.
Receives:

Fig. 13-3

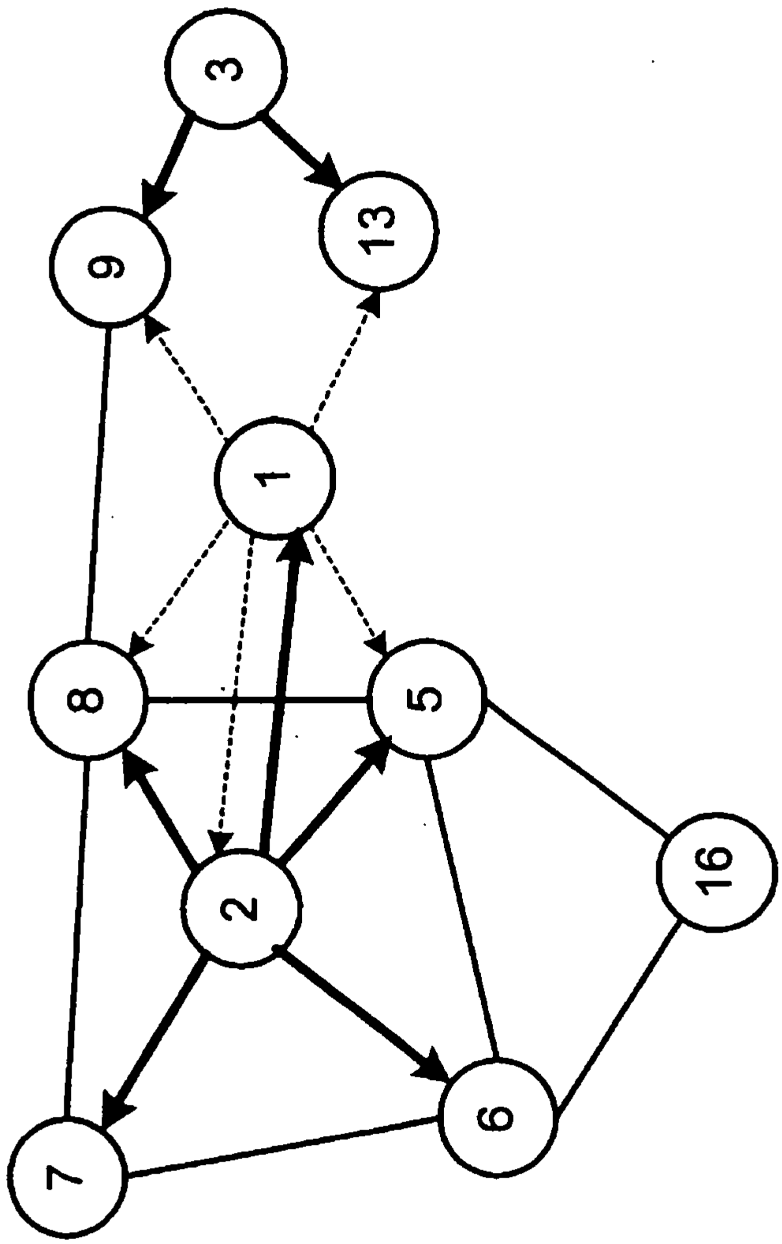


FIG. 14

UNIFYING SLOT ASSIGNMENT PROTOCOL MULTIPLE ACCESS SYSTEM

FIELD OF THE INVENTION

The present invention relates to a method for managing communication resources between nodes of a network, and more particularly to a dynamic distributed multi-channel Time Division Multiple Access (TDMA) slot assignment method.

BACKGROUND OF THE INVENTION

Mobile multi-hop broadcast packet radio networks are known for their rapid and convenient deployment, self organization, mobility, and survivability. In this type of network as illustrated in FIG. 1, a transmission from one node, for example node 1, is broadcast to all nodes in its "neighborhood". Ultra-high frequency (UHF) systems generally have a neighborhood defined by nodes within line of sight of the transmitting node (these nodes being termed within one "hop" of the transmitting node). For example, in FIG. 1 nodes 1, 3, 4, 5, 6, 7, and 8 make up one neighborhood. For data transmitted from node 1 to propagate multiple hops, the data must be relayed by one or more of node 1's neighbors. For example, node "a" (likewise nodes b, c, and g) is two hops away from the node 1 transmitter. The data will be relayed in this manner until it has arrived at all intended destination nodes.

Since there are generally limitations on the number of simultaneous transmissions that a receiver can successfully process (typically one), collisions can be avoided by the assignment of time slots in which individual nodes can transmit. There are many approaches to deciding which nodes are assigned which slots, and the approach is generally driven by the network applications, such as, broadcast, multicast, unicast, datagrams, virtual circuits, etc. Because the problem of optimally assigning slots in this environment is mathematically intractable, a heuristic approach is taken to design an integrated protocol that both chooses the number of slots to assign to each neighboring node and coordinates their activation in the network.

Tactical military and commercial applications require self-organizing, wireless networks that can operate in dynamic environments and provide peer-to-peer, multi-hop, multi-media communications. Key to this technology is the ability of neighboring nodes to transmit without interference. Neighboring nodes transmit without interference by choosing time slots and channels that do not cause collisions at the intended unicast or multicast receivers. This functionality is provided by the Unifying Slot Assignment Protocol (USAP) which is the subject of U.S. Pat. No. 5,719,868 the disclosure of which is herein incorporated by reference. The function of USAP is to monitor the RF environment and allocate the channel resources on demand and automatically detect and resolve contention resulting from changes in connectivity.

Wireless channel access schemes traditionally come in two flavors: contention and reservation. Contention has been the favorite for ad hoc broadcast networks because its lack of structure lends itself well to the mobile environment. Also, low access delays make it suitable for both tactical voice, where push-to-talk is the norm, and bursty data. However, when the network is heavily loaded contention is inefficient. At that point a structured access like reservation Time Division Multiple Access (TDMA) can achieve much higher efficiencies.

Another division in this field is the difference between broadcast and unicast. Broadcast techniques, in which a

node transmits to all of its neighbors, and unicast techniques, in which a node transmits to only one of its neighbors, both have their own unique advantages dependent upon the application. Traditional wireless systems, however, are not capable of utilizing both broadcast and unicast schemes.

Ideally, a wireless channel access system would support both reserved circuits and datagrams in whatever combination is required. Conventional wireless channel access systems however are not capable of supporting both reserved circuits and datagrams.

Conventional wireless communication systems are capable of handling only one or a small number of data types, including low latency voice, delay tolerant data, bursty transactions, high throughput streams, error sensitive data, and error tolerant video. However, conventional systems are not capable of handling the full range of data types.

Conventional wireless communications systems are typically capable of establishing and maintain only one type of virtual circuits, including either the establishment and maintenance of hard circuits or the establishment and maintenance of permanent datagrams service that can allocate soft circuits in response to traffic surges. Furthermore, conventional wireless communication systems are not able to supply selective hop-by-hop reliability to both types of traffic to overcome the effects of collisions and errors induced by the channel.

Thus, there is a need and desire for a channel access scheme to use TDMA in contention and reservation modes of operation to accommodate changing traffic patterns. Further, there is a need and desire for a wireless communication scheme capable of using both broadcast and unicast techniques dependent upon the state of the communication environments. Further still, there is a need and desire for a wireless communication system that is able to handle a full range of data types including low latency voice, delay tolerant data, bursty transactions, high throughput streams, air sensitive data, and air tolerant video. Further still, there is a need and desire for a channel access system that is capable of providing both reserve circuits and datagram service. Further still, there is a need and desire for a wireless communication system that is able to handle the full range of data types while optimizing for densities ranging from fully connected to sparse and because of its reliance only on local information is able to scale to large network sizes.

SUMMARY OF THE INVENTION

The present invention relates to a method for automatically managing the communication channel resources between two nodes having neighboring nodes in a network of transceiver nodes. Each node communicates during specific time slots and uses multiple frequencies on a time multiplex basis. The method includes storing a table of possible communication time slots and frequencies between nodes in the network at each node. The method further includes announcing and transmitting from a first node during a specific time slot, a specific transmit slot and frequency and the identification of a second node to all neighboring nodes of the first node including a first set of neighboring nodes. Further still, the method includes transmitting from the first node a control packet containing the set of transmit slot and frequency pairs on which the first node is transmitting, the set of transmit slot and frequency pairs on which the first node is receiving, and the set of transmit slot and frequency pairs on which the first set of neighboring nodes are transmitting on. Further still, the method includes identifying in the tables of each of the nodes of the first set

of neighboring nodes the announced selected transmit slot and frequency used to establish substantially contention free communication on the selected transmit slot and frequency between the first and second nodes, and adapting to the transceiver node network topology and available channel resources by utilizing a set of predetermined heuristics.

The present invention further relates to a communication network. The communication network includes a network of transceiver nodes, each node having neighbors, utilizing a time division multiple access structure, the time division multiple access structure having broadcast slots. The time division multiple access structure includes a slot assignment protocol that chooses the number of slots to assign to each neighboring node and coordinates the activation of the slots for the neighboring nodes, and at least one adaptive slot handling heuristic that controls the assignment of slots.

The present invention still further relates to a method for automatically managing the communication channel resources between nodes having neighboring nodes in a network of transceiver nodes. Each node communicates during specific time slots and uses multiple frequencies on a time multiplex basis. The method includes storing a table of communication information including frequencies, broadcast slots, and reservation slots, within a frame of a time division, for controlling and tracking communication between nodes in the network, at each node. The method further includes utilizing a time domain multiple access algorithm to control communication traffic between the network nodes, and adapting to the transceiver node network topology and available channel resources by utilizing a set of predetermined heuristics.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will become more fully understood from the following detailed description, taken in conjunction with the accompanying drawings, wherein like reference numerals refer to like parts, in which;

FIG. 1 is a general diagram of a network including nodes;

FIG. 2 is a general block diagram of the USAP multiple access system;

FIG. 3 is a diagram illustrating node activation;

FIG. 4 is a diagram illustrating link activation;

FIG. 5 is a diagram illustrating a time division multiple access structure utilized with the present invention;

FIG. 6 is a diagram illustrating the frame structure and slot schedule for the USAP multiple access system;

FIG. 7 is a diagram illustrating exemplary broadcast cycles;

FIG. 8 is a general diagram of a network depicting different broadcast cycles for different nodes;

FIG. 9 is a diagram illustrating different exemplary broadcast schedules;

FIG. 10 is a diagram illustrating an exemplary process of slot poaching;

FIG. 11 is a diagram illustrating a time division multiple access schedule using channelized neighborhoods;

FIG. 12 is a diagram illustrating an exemplary channel assignment order;

FIG. 13 is a diagram illustrating an exemplary shifting transmit schedule for each channel; and

FIG. 14 is a general diagram illustrating a network of nodes showing traffic associated therewith.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

1. USAP Fundamentals

The problem of optimally assigning slots to be practical in a wireless channel access environment is intractable. Prior to USAP a heuristic approach was typically taken to design an application specific protocol that both chose the number of time slots (hereinafter referred to simply as "slots") to assign to each neighboring node and coordinated their activation. The USAP approach separates the slot assignment mechanism from the heuristic and creates a single generalized protocol for reliably choosing the slots and coordinating their activation. This can then be used to support many different higher layer heuristics for selecting the number of slots to be activated with which neighboring nodes. Because of its inherent ability to tie together diverse higher layer protocols, it is called the "Unifying Slot Assignment Protocol".

Referring now to FIG. 2, a USAP multiple access (USAP-MA) system **10** is depicted. USAP-MA is a channel access system that results from the integration of USAP **11** with a family of heuristics **12** that enable it to satisfy the requirements of the Soldier Phone system and similar self-organizing wireless networks operating in a dynamic environment and providing peer to peer, multi-hop, multi-media communications. USAP-MA supports a collection of requirements in a flexible fashion so that the best instantaneous mix can be achieved to match the applications, the loading, and the topology.

Traditionally, there are two methods in which a node transmits to its neighbors, depicted in FIGS. **3** and **4**. As depicted in FIG. **3**, node activation describes the technique in which a single transmitting node (here depicted as node #2) broadcasts to all of its neighbors simultaneously rather than on an individual basis. When a transmitter has only one intended receiver (for example transmitter #2 has a single receiver #1 depicted in FIG. **4**) it is known as "link activation". The former allows only one active transmitter in a neighborhood while the latter can potentially have several.

Node activation is especially well suited for applications like address resolution and conferencing. Link activation, on the other hand, lends itself better to high volume point-to-point traffic. As depicted in FIG. **2**, USAP-MA **10** has heuristics to support both under the more common names of multicast **15** (or broadcast) and unicast **20**.

Hereinafter, an ordered pair of RF channel and time slot may be referred to as an "allocation". When a node chooses an allocation, USAP enforces certain constraints to avoid interference within 2 hops of the transmitter. For unicast from node i to neighbor j it must be an allocation:

that has not already been assigned to either node;

i's neighbors are not receiving in; and

j's neighbors are not transmitting in.

For multicast a node i must choose an allocation:

that has not already been assigned to node i or any of its neighbors; and

none of i's neighbors' neighbors are transmitting in.

A node insures that its allocations satisfy the above constraints by sharing the following USAP slot sets with its neighbors:

STi—allocations where a node is transmitting;

SRi—allocations where a node is receiving; and

NTi—allocations where a node's neighbors are transmitting.

To minimize the size of the control packets needed to share this information, it may be more efficient to encode

these sets as bit maps or as lists, depending on the number of slots and channels being managed and the density of the network. By exchanging these control packets, USAP has the information it needs to choose non-conflicting transmit allocations consistent with the most recent topology measurements and detect and report conflicts caused by topology changes.

After a transmit allocation is chosen, a node has the option of either using it immediately or waiting until a confirmation is received from each neighbor. The unconfirmed mode is appropriate when it is tolerable to have momentary conflicts due to coincident changes in connectivity or conflicting allocations. The confirmed mode verifies that all neighbors are aware of the allocation and that nothing has happened in the meantime to make it inconsistent with the current topology or other nodes' allocations.

To decide if a slot and a channel are available for either broadcast or unicast transmissions, the USAP methodology may be applied, which is the subject of U.S. Pat. No. 5,719,868 and is described below, however other alternative methodologies exist and could be used in lieu of the USAP methodology.

a. USAP Slot Sets

To choose a slot, a node first generates the set of slots that are not available because they are already in use locally. In the description that follows, the subscript of "i" denotes information about this node and "j" denotes the corresponding information reported by a neighboring node.

Slot Assignments:

S=set of slots

F=set of channels

Self Transmit/Receive Sets:

STNi(s,f)=set of neighbors to which node i transmits on (s,f)

SRNi(s,f)=set of neighbors from which node i receives on (s,f) from which the following sets are derived:

STi(s,f)=1 if STNi(s,f) not empty, else 0

SRi(s,f)=1 if SRNi(s,f) not empty, else 0

Neighbor Transmit/Receive Sets:

STj(s,f)=the STi(s,f) reported by neighbor j

SRj(s,f)=the SRi(s,f) reported by neighbor j from which the following sets are derived:

$$NTi(s,f) = \bigcup STj(s,f) \text{ over all neighbors } j \text{ of node } i$$

$$NRi(s,f) = \bigcup SRj(s,f) \text{ over all neighbors } j \text{ of node } i$$

NTj(s,f)=the NTi(s,f) reported by neighbor j

If node i or its neighbor j is transmitting or receiving (on any channel) in slot s, it is blocked from doing anything else. To this end the following derived sets are useful:

$$Bi(s) = STi(s,f) \cup SRi(s,f) \text{ for any } f$$

$$Bj(s) = STj(s,f) \cup SRj(s,f) \text{ for any } f$$

To decide what slots and channels are available for unicast allocation, a node i constructs the blocked allocations for transmitting to j by excluding allocations

already assigned to either node:	$Bi(s) \cup Bj(s)$
i's neighbors are receiving in:	$NRi(s,f)$
j's neighbors are transmitting in:	$NTj(s,f)$

This information is combined as follows:

$$\text{Blocked}(i,j,s,f) = Bi(s) \cup Bj(s) \cup NRi(s,f) \cup NTj(s,f)$$
$$= 1 \text{ if } i \text{ cannot transmit to } j \text{ in } (s,f), \text{ else } 0$$

To decide what slots and channels are available for broadcast allocation, a node i constructs the blocked allocations for transmitting to all of its neighbors by excluding allocations

already assigned to i:	$Bi(s)$
already assigned to any of its neighbors:	$\bigcup_{\forall n \in \{i's\_nbrs\}} Bj(s)$
any of i's neighbors' neighbors are transmitting in:	$\bigcup_{\forall n \in \{i's\_nbrs\}} NTj(s,f)$

This information is combined as follows:

$$\text{Blocked}(i,s,f) = Bi(s) \bigcup_{\forall n \in \{i's\_nbrs\}} Bj(s) \bigcup_{\forall n \in \{i's\_nbrs\}} NTj(s,f)$$
$$= 1 \text{ if } i \text{ cannot transmit to any of its neighbors in } (s,f), \text{ else } 0$$

b. Reducing Receiver Noise Floor

The data structures of the previous sections are designed to prevent conflicts at a receiver due to multiple transmitters within its neighborhood using the same frequency in the same slot. In some networks it may be desirable to reduce the noise floor at the receivers by adding an additional hop of isolation before an allocation is reassigned. This can be accomplished by defining the set of nodes that are transmitting within 3 hops of node i on an allocation as

$$NNTi(s,f) = \bigcup NTj(s,f) \text{ over all neighbors of node } i$$

If this is also included in the control packet and stored at the neighbor of node i as NNTj(s,f) and then the Blocked (i,j,s,f) generated as follows will prevent conflicting transmissions within 3 hops of a transmitter:

$$\text{Blocked}(i,j,s,f) = Bi(s) \cup Bj(s) \cup NRi(s,f) \cup NTj(s,f) \cup NNTj(s,f)$$

$$= 1 \text{ if } i \text{ cannot transmit to } j \text{ in } (s,f), \text{ else } 0$$

2. TDMA Structure

The objectives of many wireless channel access systems are to support both datagrams and on demand circuits, low latency voice, and network size, density, and mobility consistent with the tactical battlefield. After considering a number of tradeoffs, USAP-MA may have, in a preferred embodiment the TDMA structure depicted in FIG. 5, although other structures are possible and, given other constraints, more optimal.

As depicted in FIG. 5, the time division multiple access structure can be viewed as three individual cycles, a bootstrap cycle **25**, a broadcast cycle **30**, and a reservation cycle **35**. Each of the cycles depicted are, in a preferred embodiment, a one second cycle, each of the cycles being divided into eight even frames, each frame being 125 milliseconds in length. (The number of cycles, the length of the cycle, and the length of each frame is a matter of design choice, and should not be viewed as limiting; other numbers of cycles, frame lengths, and cycle lengths may be used without departing from the spirit and scope of the present invention.) Each 125 millisecond frame includes a number of bootstrap minislots **40**, broadcast slots **45**, and reservation/standby slots **50**. Further, each frame includes a plurality of frequencies, shown as channels **55**.

The frame length of 125 milliseconds is practical because it is deemed acceptable for voice latency, it divides 1 second evenly, and it allows a manageable number of slots per frame. Bootstrap minislots **40** are used for sharing the critical USAP information necessary to dynamically assign the rest of the slots. With 13 bootstrap slots per frame and a bootstrap cycle of 4 frames, up to 52 nodes can be supported although a technique for managing larger nets may also be incorporated into the system. Broadcast slots **45** are node allocations made to support the datagram service and any other control traffic that nodes need to share. With a broadcast cycle of 4 frames, up to 8 nodes can transmit in the 2 broadcast slots or a total of up to 40 nodes on 5 different channels in a neighborhood. Reservation slots **50** repeat every frame so their latency is 125 milliseconds. When a reservation slot has not been allocated it acts as a standby broadcast slot (controlled by standby slot heuristic **126**, FIG. 2) assigned to the same transmitter as the corresponding broadcast slot. In addition, the correspondence is shifted by 1 slot every frame as depicted in FIG. 6 (each standby broadcast slot being indicated by the labels (B0) through (B7)), to minimize the impact on the broadcast capacity and latency of any one node when a reservation slot is allocated.

For example, FIG. 6 depicts eight frames representative of a single channel over a one second period. Each frame has eight reservation/standby broadcast slots **50**. Each of slots **50** may be either a broadcast slot, represented by (B0) through (B7), or a reservation slot represented by R0 through R7. In the first frame, (B0) is in the first column, in the second frame (B0) is shifted by one slot to be in the eighth column, in the third frame, F2, (B0) is in the seventh column, and so on until in the eighth frame, F7, B0 is in the second column.

The upshot of the TDMA schedule, depicted in FIG. 6, is that when no reservations are active all slots are fully utilized in support of a broadcast channel that can be used for user datagrams or control traffic. As unicast reservations become active they gradually decrease the capacity of the broadcast channel. Note however that a broadcast reservation actually increases the broadcast capacity of the transmitting node.

3. Bootstrap Slots

For small networks it is most efficient to assign one permanent bootstrap slot per node, but as network size increases this method results in less and less traffic capacity. Thus, it is desirable to have a technique for dynamically assigning these slots (i.e., bootstrap slot heuristic **128**, FIG. 2). However, since the bootstrap packets carry the USAP information necessary for contention-free assignment, each node needs initialization information to coordinate itself with its neighbors. Each new or entering node listens for bootstraps from its neighbors before assigning itself a slot and transmitting its own bootstraps. If a conflict occurs, the effected nodes will learn this from their neighboring nodes via the USAP information and choose another slot.

A fixed or dynamic bootstrap assignment system can be used if a dynamic bootstrap assignment method is used, the bootstrap cycle length will depend on the maximum expected neighborhood density. Assuming a network of size N with a maximum degree D (the most neighbors any one node can have), the following formula can be used:

$$L=\min(D^2+1,N)$$

The D^2+1 term is a simple upper bound on the number of colors it takes to do a distance-2 vertex coloring of an undirected graph and this corresponds to the maximum number of bootstrap slots a neighborhood needs for every node to get a slot. (Thus, other formulations for L may be applied without departing from the spirit and scope of the present invention.) One can see that small and/or fully connected networks are best handled by fixed assignments. However, for large networks of reasonable degree the D^2+1 term yields a smaller cycle. Indeed, this demonstrates that USAP-MA can manage very large networks with a resource management latency that depends only on the maximum neighborhood density. In fact, that latency can be improved significantly for sparser neighborhoods by simply assigning more slots to each node as described below.

An efficient but flexible way to assign more or less slots is to do it in blocks whose size is a power of 2 and whose slots are evenly distributed in the bootstrap cycle. This allows the block to be specified by the first slot index and N where 2^N is the size of the block. In fact, if the first slot indices are chosen to be as small as possible then the size of the blocks can be adjusted up or down with maximum flexibility. This technique is very similar to the Adaptive Broadcast Cycle and Slot Poaching to be described shortly but it deserves its own treatment here. Consider the short broadcast cycle **60** depicted in FIG. 7 that can accommodate up to 8 nodes.

Sparser neighborhoods could optimize down to 4 node cycles **65** or 2 node cycles **70** if the nodes continually assign themselves the lowest available slot index, thereby packing themselves into the lower portion of the cycle. Under certain conditions a node can reuse unassigned slots in the latter part of the cycle if they correspond to its transmit slot on a shorter

cycle. A node gleans this knowledge about its neighbors from the USAP information in their bootstraps, as will be described with regards to slot poaching. An exemplary network where this optimization is possible is depicted in FIG. 8.

The nodes in FIG. 8 are numbered by their slot index and the first group 75 has packed themselves into an 8-node cycle. This allows the 3 nodes of second group 50 the right to use a 4-node cycle. It should be noted that the present invention is in no way limited to the configuration or the node cycles depicted in FIG. 8. Any of numerous configurations and node cycles could be used while still achieving a methodology for dynamically assigning bootstrap cycles.

4. Adapting to Network Density

One of the hardest problems to solve in broadcast networks is how to efficiently manage network densities ranging from sparse to fully connected. USAP-MA uses a combination of techniques to solve this problem. To handle sparse neighborhoods the Adaptive Broadcast Cycle 81 (ABC), depicted in FIG. 2, employs nested subdivisions of the fixed length cycle to allow nodes to reuse unused slots. On the other hand, neighborhoods whose density exceeds the capacity of the broadcast cycle utilize Channelized Neighborhoods 90 (CN) to create parallel cycles on other channels so that multiple neighboring transmitters can be active in any given slot. The fusion of these methods results in efficient channel utilization over the entire range of connectivity.

a. Adaptive Broadcast Cycle (ABC)

A number of possible broadcast schedules are depicted in FIG. 9, each for a different neighborhood density, that is, the maximum number of nodes before CN 90 (FIG. 1) takes effect. As depicted in FIG. 9, a number in a slot corresponds to the assigned broadcast slot. In other words, if a node has chosen broadcast slot N, the standby slots with the number N are also it's transmit opportunities. They are designed to have the following properties:

- They maximize overlap such that a schedule has half of its slots in common with both the previous schedule (shown shaded) and the next schedule. This eases transitions from one schedule to the next as the density changes.
- The standby slots are shifted from frame to frame to minimize the impact a reservation slot has on broadcast latency.

A schedule is chosen based on the location of the assigned broadcast slot and any open slots that come after it in the broadcast cycle. If there are open slots corresponding to the assigned slot on a shorter cycle, the node may be able to adopt the shorter cycle as described with regards to slot poaching, described below. To make this work a node continually looks for empty slots earlier in the cycle and tries to move its broadcast allocation to that slot. This has the effect of packing the allocated slots into the earlier part of the cycle and allowing the cycle to be shortened as density decreases.

b. Slot Poaching

One way to implement ABC 81 (FIG. 1) is to allow a node to transmit in or "poach" unassigned slots according to a shorter cycle if no other transmitter will use them within two hops of this node. This feature exploits the USAP information available at each node, the schedules of nested blocks of 2^k slots, and the fact that a node can be ready to receive in any slot. If the basic broadcast cycle is long enough to accommodate the maximum network density, poaching will provide shorter cycles for sparser neighborhoods. For instance, poaching would allow an 8-node schedule to achieve the efficiency of a 2 or 4 node schedule whenever possible.

The set of slots P consisting of the original assigned broadcast slot plus the poached slots can be defined as

$$P=\{p|p=i+j2^k\}$$

Where i=index of assigned broadcast slot

j=0, 1, . . . , n such that p<basic broadcast cycle

k=integer such that 2^k=size of cycle and $i<2^k$

In order for P to be free of conflicts, the intersection of it with Blocked(i,s,f) must be empty except for the assigned broadcast slot i. This can be accomplished with the following procedure:

1. Start with the smallest k that satisfies $i<2^k$.
2. If 2^k is the basic broadcast cycle then no poaching is possible and we are done.
3. If the intersection of P and Blocked(i,s,f), from USAP, is just i, then P is valid and we are done. Else, increment k and go to 2.

In FIG. 10, an 8 slot basic cycle is represented with the blocked slots marked with a 1. The nodes transmitting in the left half of the cycle try to poach. Slot 82 can poach slot 83 and slot 84 can poach slots 85, 86, and 87. However, slot 88 is blocked by slot 89 and cannot poach. Note however that if slot 88 were to move to slot 85 (depicted in the second cycle 91) then it could poach also, albeit at slot 84's expense. This shows that the packing in the lower part of the cycle can be optimized to maximize the potential for poaching.

A simple proof that this node will not conflict with another node trying to poach the same slot is as follows. Step 1 establishes the smallest poach cycle this node will consider while step 3 increases the size of the cycle if another transmitter is active in P. If there is another active transmitter, it will have an i greater than this node by a factor of 2^k, say $m2^k$, and have a P defined as follows with respect to this node:

$$P=\{p|p=i+m2^k+r2^q\}=\{p|p=i+(m+r2^{q-k})2^k\}$$

Where i=index of assigned broadcast slot of this node

r=0, 1, . . . , n such that p<basic broadcast cycle

q=integer such that 2^q=size of cycle and $(i+m2^k)<2^q$

Thus, this node will not consider an unused slot without also considering the slot assigned to the other transmitter (r=0) that might also consider that slot, which will qualify P. Thus, two nodes can not choose the same P.

c. Offline Neighborhoods

Nominally ABC 81 utilizes just one channel. However, if a group of nodes needs to share data at a higher rate than is provided by the datagram service of the broadcast channel, they may utilize one of the other available channels and take advantage of a shorter broadcast cycle to create additional throughput. For instance, a videoconference could be set up that might tax the normal broadcast channel. However, if there were only a few nodes involved and they could afford to temporarily form their own subnet, they could switch to another channel and use a shorter broadcast schedule. Note that they would still maintain contact with the main network via the bootstrap packets in case of an urgent request to return to the main channel. This would not only allow them to greatly increase the throughput among themselves but also, since there are fewer nodes involved, to minimize the control traffic that they must share to maintain the subnet.

5. Channelized Neighborhoods

If the number of broadcast slots is limited to 8, communication with more than 7 neighbors requires the use of additional broadcast cycles, where cycles can be separated by time or channel that is into channelized neighborhood 90

(FIG. 2). USAP-MA uses the latter approach to maximize channel utilization. For a receiver to hear from each transmitter, it rotates from one channel to another according to the schedule in which a 2-second epoch is broken into 4 cycles, each of 4 frames. As depicted in FIG. 11, the schedules for the 8, 16, 24, and 32 node cases are presented. A Rotating Receiving Group (RRG) is made up of the 4 pairs of nodes that share the same frame on each channel. An RRG will visit a different channel each frame so that at the end of an epoch (in a preferred embodiment an epoch lasts 2 seconds however, an epoch may be of any defined length) each receiver will get a chance to receive from up to 28 transmitters in a 32-node neighborhood. The reason that not all 31 transmitters are heard at every receiver is that the nodes with the same transmit slot will never receive from each other. This has the result that these nodes will not be recognized as logical neighbors even if they are physical neighbors. Communication between them can be achieved over relays or a further optimization can be made to change the RRG memberships each epoch in such a way as to evenly distribute the receive opportunities over time.

The {T0,T1,T2,T3} schedule **100** presented for the 32-node neighborhood can be optimized for sparser neighborhoods. In particular, an 8-node neighborhood uses a {T0,T0,T0,T0} schedule **105** for single channel operation, a 16-node neighborhood uses a {T0,T1,T0,T1} schedule **110** for 2-channel operation, and a 24-node neighborhood uses a {T0,T1,T2,T0} schedule **115** for 3-channel operation. Since the perception of neighborhood density can change from one node to the next, the nodes must let each other know each other's schedules. This is done via a couple of bits in the bootstrap. When a node goes to transmit, it can calculate which of its neighbors will be listening at that time according to the schedule in effect at each neighbor. Note that each node always transmits in its allotted time slot and channel. The receivers are the ones that rotate among the transmitters on different schedules depending on the neighborhood density and the RRG they belong to.

a. Channel Assignment Order

USAP-MA uses a further optimization to decrease the impact on the broadcast channel of assigning unicast reservation slots. As depicted in FIG. 12, the USAP-MA has, in a preferred embodiment, 5 channels available to it numbered 0 to 4 (labeled C0 to C4 in FIG. 12). If the broadcast channels are assigned in the order of 0 to 4 and the reservation slots on channels in the reverse order of 4 to 0, there will be minimal overlap between the broadcast and reservation slots until the network gets heavily loaded. In fact, when a unicast reservation is made on channel 4, it has no impact on the broadcast channel except that the two unicast nodes are unavailable during that slot for broadcast. Furthermore, if reservations were limited to two channels, which still gives much flexibility for unicast assignments, there would be no possibility of reduced broadcast capacity until the neighborhood density reached 24 and the broadcast channel spilled onto channel 3. This is yet another example of how USAP-MA maximizes channel utilization while allowing substantially simultaneous contentionless unicast and broadcast transmissions in the same neighborhood.

b. Fair Channelized Schedules

In order to allow all nodes to receive from each transmitter in a fair fashion, it is necessary to periodically change the RRG membership. Recall that the RRG is made up of the 4 pairs of nodes that share the same frame, where one pair comes from each channel. By shifting the transmit schedule on each channel a different amount each 2 second epoch a different set of receivers will be active on any particular slot and channel from one epoch to the next. One such shift set {0,1,3,5} results in the following shift schedule for each channel over a 16-second period:

Chan	Epoch 0	Epoch 1	Epoch 2	Epoch 3	Epoch 4	Epoch 5	Epoch 6	Epoch 7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	3	6	1	4	7	2	5
3	0	5	2	7	4	1	6	3

The first three epochs produced by this shift schedule are depicted in FIG. 13, where the active receivers for broadcast slot 0 channel 0 are shown shaded with the degree of shade representing the number of active receivers. Note that in this diagram the times of the receptions are significant only with respect to the frame they occur in. Also, the slot numbers correspond to the assigned broadcast slot. As expected, for the 20 transmissions that each node has per period there are 140 receptions or 7 receivers at a time on each channel. It is plain to see that the distribution of receptions is different for every cycle. What is not so evident is that over the 16-second period each node has an equal number of opportunities to receive each transmitter, but this can be shown with the following analysis:

1. In every 4-frame cycle all nodes have an equal opportunity to receive on all channels. This is because during this time each RRG visits each channel once. Thus, during the 16-second period all nodes visit each channel equally.
2. Since in every 2-second epoch all receivers are active except while transmitting, the transmissions they will miss are the ones either being transmitted on a channel other than the one they are listening to or while they themselves are transmitting. But step 1 shows that they will miss an equal number on all channels, so we must only account for the ones missed while each node is transmitting. In other words, if each transmitter has an equal opportunity of missing transmissions from all other simultaneous transmitters, then the receive opportunities are equal.
3. Now, in any particular 2-second epoch the simultaneous transmitting nodes on each channel have a relationship specified by the shift schedule of the above table. The table itself corresponds to broadcast slot 0 but there is a similar schedule for the other 7 slots. If we can show that each of the simultaneous transmitters for slot 0 misses the same number of transmissions from slot 0 over the 16-second period, then by step 2 the receive opportunities are equal.
4. It is a simple matter of counting in the table to see that each slot aligns with slot 0 exactly 3 times per 16-second period so by step 3 the receive opportunities are equal.

One other optimization can be made to minimize the latency that is caused by the clumping of slots each epoch. For example, epoch 0 has 4 receivers in only one or two slots per frame resulting in a latency of a half-second or more between receives opportunities at any particular node. Since other epochs have a more even distribution of receivers per frame worst case latency could be reduced by methodically swapping frames from high latency epochs with corresponding frames in low latency epochs.

6. Error Detection and Retransmission

Some types of traffic benefit from reliable hop-by-hop transmissions. That is, if incorrectly received packets and collisions are detected, the affected packets can be retransmitted. This is achieved in USAP-MA 10 with a bitmap in the bootstrap packet representing each slot in the frame where each bit means that a packet was correctly received in the corresponding slot. If a node had previously transmitted in a slot then it can tell if its intended receiver(s) correctly received the packet and if not then retransmit it. This technique applies equally well to transmissions in the broadcast, standby, and reservation slots. Of course, some types of traffic like voice do not benefit from retransmissions so this facility can be applied selectively.

7. Contention Access or Spec Slots

There are circumstances when traffic is sporadic and bursty. Traditionally this is most efficiently handled by contention access while reservation access is most efficient under heavy loads. Now, in the absence of allocations USAP-MA 10 effectively reserves all slots, either permanently or temporarily, for neighbors to broadcast in. However, under conditions where contention access is desirable, it would be more efficient to allow nodes to transmit in standby slots that do not belong to them but that are going to be idle for lack of traffic. In other words, if a node knows that in the upcoming frame it is not going to need its standby slot, it could announce this fact in the bootstrap packet (StandbyFree) at the beginning of the frame so that another node could transmit in that slot if needed. The other node can then determine (1) if it can effectively use the slot and (2) if there is a good chance that the intended receiver(s) will hear its transmission, hence the term "Spec (ulation) Slot" 130 (FIG. 2). Since there is contention for the slot, there may be collisions but these can be compensated for using the error detection already described.

A node receiving a bootstrap with StandbyFree set would consider using this neighbor's standby slot if it currently had more traffic than it could send in its reserved slots in this frame. If so, it would determine if any of its queued traffic were addressed to neighbors who would normally be listening on the right channel. This would include the original neighbor and any other neighbor that it and this node have in common. Now, if any of these common neighbors happen to also use the slot speculatively then there may be a collision, depending on the location of the intended receivers with respect to the transmitters. To determine if any of its neighbors are listening on the right channel a node consults the USAP self-receive tables (SRj) for each neighbor. Alternatively, a node could more accurately calculate the chances of collision if the bootstrap Assigned Slot Record (ASR) also contained NRi, which would yield whether a neighbor is allowed to transmit.) Furthermore, to determine if these neighbors are in common with the original transmitting node, a node will need to consult its routing information. Once qualified neighbors are identified, traffic for them can be queued up to go out in the slot.

As an example, consider that node 2 in FIG. 14 might announce StandbyFree as set. If node 1 had excess traffic that it wished to send to nodes 2, 5, and/or 8 it could use node 2's standby slot as a spec slot, assuming that node 3 was not transmitting at the same time on the same channel. Otherwise there would be a collision at nodes 9 and 13. This is the reason for checking that all of the potential receivers of 1's speculative transmission are neighbors of node 2. Since this information is normally not stored by USAP it must be obtained elsewhere, possibly from the routing protocol.

8. Bootstrap Packet Format

The bootstrap packet consists almost entirely of the USAP assigned slot sets STi, SRi, and NTi. The broadcast and reservation slots are managed separately and encoded slightly differently. There is a map of 2 bit fields that correspond to every slot, yielding 2 bits*8 slots*5 channels=80 bits for the reservation slots and 2 bits*8 slots=16 bits for the broadcast slots. The 2 bit fields are combined into what is called the Assigned Slots Record (ASR) and are interpreted as follows:

Broad-cast:	0: unassigned	Reservation:	0: unassigned
	1: self transmit		1: self transmit
	2: self receive		2: self receive
	3: conflict		3: neighbor transmit or conflict

USAP regenerates the STi, SRi, and NTi sets from the ASR when it receives a bootstrap packet from a neighbor. In addition, there is

1 bit for StandbyFree;

1 bit for USAP confirmation;

2 bits to indicate which of 4 channels the broadcast slots are used on;

2 bits for channelized neighborhood cycle (8, 16, 24, or 32-node);

4 bits for 16 second synchronization;

24 bits to identify up to 4 6 bit receiver Ids for unicast reservations; and

10 bits for error detection.

In this way the entire functionality of USAP-MA 10 described up to this point is packed in about 18 bytes. Alternatively, this 18 byte packet may be of any size that is suitable for the given resources and should not be considered limiting.

9. Neighbor Segregation

A problem with adapting transmission parameters on a broadcast channel is that a node may have to use parameters that work with the worst neighbor to insure that all nodes receive the broadcast. A worst cast scenario is when a node has a set of good quality neighbors that it would normally be able to transmit to at a high rate but it is stuck using a low rate to achieve reliable transmissions to a single poor neighbor. The USAP-MA 10 solution is to segregate neighbors (controlled by neighbor segregation heuristic 132, FIG. 2) into 2 (or more) groups and adjust the transmission parameters to each of them independently. In this way a node could transmit to its good neighbors at an efficient high rate and its poor neighbors at a robust lower rate.

Of course, in order to use two different rates, a node either needs to have different transmission opportunities or be able to change rates in the middle of a packet. Between the broadcast and standby slots a node nominally has two broadcast opportunities. Since the standby slot has lower latency and the potential for higher throughput it makes sense to use it for the high quality neighbors and use the broadcast slot for the poorer neighbors. Of course, if all

neighbors are equally good then both slots can be used interchangeably.

Packets that must go to all neighbors (like routing updates) or to one or more of the poor neighbors would thus use the broadcast slot. Packets destined for the good neighbors would use the standby slot unless there was room in the broadcast slot left over. Note that the routing should tend to favor the good neighbors, thereby directing most of the packets into the higher rate standby slot.

Besides the increased efficiency of transmitting at higher rates, neighbor segregation also allows the receivers to sample a greater number of data rates, enabling them to better choose the proper transmission parameters for their neighbors.

10. Soft Circuits

One of USAP's strong points is its ability to make allocations on the fly. Since it constantly maintains up-to-date knowledge via the bootstrap packets of the active allocations its it neighborhood, it is always ready to allocate a new transmit slot. Nominally the new allocation must be announced in only one bootstrap before it can be used. Certain types of applications can take advantage of this to eliminate the setup delays and inflexibility of hard circuits **134** (FIG. 2), that is, reserving slots all along a path from source to destination before turning on the traffic. In particular consider voice and video. Once started they continuously supply data that loses its value if it is not delivered very quickly. They are also both tolerant of small amounts of dropped data and data arriving out of order, resulting in small glitches of little consequence to the users.

Now consider a heuristic that simply assigns enough slots on its outgoing links to satisfy the requirements of the traffic entering it. This would allow traffic to push its way through the network reserving the channel resources along the way. When the traffic stops the allocations are automatically released. Called a "soft circuit" **136** (FIG. 2), this heuristic has essentially the same effect as a dedicated circuit setup: if enough slots are available, they are allocated to support the traffic stream. The difference is that it is executed locally at each node rather than coordinated at the end points. Also, the soft circuit heuristic will attempt to fix a broken link locally by routing around it if possible instead of reporting the failure all the way back to the source. Thus, by taking advantage of certain characteristics of the data streams and effectively utilizing the capabilities of USAP **11**, soft circuits provide fast, robust support for certain applications.

USAP **11** can manage the resource allocations for many types of TDMA architectures that operate in a distributed dynamic environment. The user needs and application requirements drive the design of heuristics to cause USAP to assign the right mix of permanent or on demand allocations. USAP-MA is an example of such an integrated family of heuristics that supports the requirements of the tactical military environment. It does this by providing broadcast and unicast transmissions, datagrams for bursty data and hard and soft circuits for high capacity or low latency streams, a seamless transition between reservation and contention access, selective hop-by-hop reliability to minimize end-to-end retransmissions, sparse and dense neighborhood optimizations, and the ability to scale to large networks.

It is understood that while the detailed drawings and examples given describe preferred exemplary embodiments of the present invention, they are for the purposes of illustration only. The method and apparatus of the invention is not limited to the precise details and conditions disclosed. For example, it is not limited to the specific time frame and time slot lengths, or to the number of cycles, or to the inclusion of all of the heuristics discussed. Various changes may be made to the details disclosed without departing from the spirit of the invention, which is defined by the following claims.

What is claimed is:

1. A method for automatically managing communication channel resources between nodes having neighboring nodes in a network of transceiver nodes, wherein each node communicates during specific time slots and uses multiple frequencies using a time division multiple access structure, the method comprising:

storing possible communications time slots and frequencies between nodes in the network at each node;

announcing and transmitting from a first node during a specific time slot, a specific transmit slot and frequency and the identification of a second node to all neighboring nodes of the first node comprising a first set of neighboring nodes;

transmitting from the first node in a bootstrap slot a control packet containing:

a set of slot and frequency pairs on which the first node is transmitting;

a set of slot and frequency pairs on which the first node is receiving; and

a set of slot and frequency pairs on which the first set of neighboring nodes are transmitting on;

identifying in each of the nodes of the first set of neighboring nodes the announced selected transmit slot and frequency used to establish substantially contention free communication on the selected transmit slot and frequency between the first and second nodes; and

assigning slot and frequency pairs from the communications channel resources by utilizing a heuristics.

2. The method of claim 1 further comprising:

transmitting from the first set of neighboring nodes to all neighboring nodes of each node in the first set of neighboring nodes comprising a second set of neighboring nodes, the announced transmit slot and frequency used to establish contention free communication between the first and second nodes, and identifying in tables of each of the second set of neighboring nodes the announced transmit slot and frequency.

3. The method of claim 2 further comprising:

the first set of neighboring nodes each receiving confirmation from a second set of neighboring nodes that the announced slot and frequency have been identified in the tables of the second set of neighboring nodes; and the first node receiving confirmation from the first set of neighboring nodes that the announced transmit slot and frequency have been identified in the second set of neighboring nodes.

4. The method of claim 3 further comprising:

transmitting a confirmed message from the second node to the first node thereby establishing communication using the selected transmit slot and frequency.

5. The method of claim 2 further comprising:

identifying, to the first set of neighboring nodes, slot and frequencies not being used by the first node to thereby make available unused slots and frequencies to the first and second sets of neighboring nodes.

6. The method of claim 1 further comprising:

resolving conflicts between nodes which have announced a desire to transmit on a same slot and frequency.

7. The method of claim 1 wherein storing possible communications slots and frequencies further comprises assign-

ing a set of bootstrap slots for all nodes to transmit the control packets.

8. The method of claim 7 wherein assigning slot and frequency pairs further includes dynamically assigning the set of bootstrap slots based on the density of neighboring nodes with a bootstrap slot heuristic.

9. The method of claim 1 wherein storing possible communication time slots and frequencies includes assigning a plurality of broadcast slots to nodes to transmit to all neighboring nodes and assigning a plurality of reservation slots to nodes to communicate between two nodes.

10. The method of claim 9 wherein assigning the plurality of reservation slots further includes assigning the reservation slots as standby broadcast slots for use by a same node as the reservation slot assignment by using a standby slot heuristic.

11. The method of claim 10 further comprising adaptively assigning the standby broadcast slots according to network density using an adaptive broadcast cycle heuristic.

12. The method of claim 11 further comprising reassigning standby broadcast slots with the adaptive broadcast cycle heuristic by using slot poaching.

13. The method of claim 9 further comprising establishing channelized neighborhoods with a channelized neighborhood heuristic that allows standby broadcast slots to be assigned on different frequencies.

14. The method of claim 10 further comprising selectively permitting contention access with a speculation slot heuristic that allows a node to transmit in an idle standby broadcast slot not assigned to the node.

15. The method of claim 11 further comprising selectively permitting contention access with a speculation slot heuristic that allows a node to transmit in an idle standby broadcast slot not assigned to the node.

16. The method of claim 1 wherein assigning slot and frequency pairs further includes segregating neighbors based upon transmission rate between a node and it neighbors with a neighbor segregation heuristic.

17. The method of claim 9 further comprising segregating neighbors based upon transmission rate between a node and it neighbors with a neighbor segregation heuristic.

18. The method of claim 1 wherein assigning slot and frequency pairs further includes dynamically assigning enough slots between nodes to satisfying traffic requirements by establishing soft circuits with a soft circuit heuristic.

19. The method of claim 1 wherein assigning slot and frequency pairs further includes reserving all slots along a path from source to destination before transmitting traffic by establishing hard circuits with a hard circuit heuristic.

20. A communication network comprising:

a network of transceiver nodes each node having neighbors;

a time division multiple access structure having broadcast slots and reservation slots and the time division multiple access structure including:

a slot assignment protocol that allows nodes to choose available transmit slots based on slot usage by neighboring nodes and coordinates the activation of the slots for the nodes;

a plurality of bootstrap slots for each node to transmit slot assignment information from the slot assignment protocol; and

at least one slot handling heuristic that adaptively and dynamically controls the assignment of slots.

21. The communications network of claim 20 wherein the plurality of bootstraps slots are fixed bootstrap slots assigned with one bootstrap slot for each transceiver node.

22. The communications network of claim 20 wherein the time division multiple access structure reservation slots are also assigned as standby broadcast slots for use by a same node as the reservation slots by using a standby slot heuristic as the slot handling heuristic.

23. The communications network of claim 20 wherein the at least one slot handling heuristic includes a bootstrap slot heuristic for dynamically assigning the bootstrap slots to the transceiver nodes.

24. The communication network of claim 21 wherein the bootstrap slot protocol is a fixed bootstrap slot protocol.

25. The communications network of claim 22 wherein the at least one slot handling heuristic includes an adaptive broadcast cycle for adaptively assigning the standby broadcast slots according to network density.

26. The communications network of claim 22 wherein the at least one slot handling heuristic includes a channelized neighborhood heuristic that allows standby broadcast slots to be assigned on different frequencies when the number of neighboring nodes increases.

27. The communication network of claim 22 wherein the at least one adaptive slot handling heuristic includes a soft circuit heuristic for allocating time slots in response to traffic demands during transmission of information.

28. The communication network of claim 22 wherein the at least one adaptive slot handling heuristic includes a hard circuit heuristic for allocating time slots prior to transmitting information.

29. The communications network of claim 22 wherein the at least one slot handling heuristic includes a speculation slot heuristic that allows other nodes to use standby broadcast slots that are unused by the node assigned to the standby broadcast slot.

30. The communication network of claim 20 wherein the at least one adaptive slot handling heuristic includes a neighbor segregation heuristic that divides the neighbors of a node into at least two groups and adjusts transmission parameters separately for each group.

31. A method for automatically managing communication channel resources between nodes having neighboring nodes in a network of transceiver nodes, wherein each node communicates during specific time slots and uses multiple frequencies using a time division multiple access structure, the method comprising:

assigning broadcast and reservation slots and frequencies to the nodes with a slot assignment protocol;

assigning bootstrap slots to the nodes;

sharing the broadcast and reservation slot and frequency assignments between nodes using the bootstrap slots; and

dynamically adapting the slot and frequency assignments by utilizing a family of heuristics.

32. The method of claim 31 further comprising

dynamically assigning the bootstrap slots to nodes of the network using a bootstrap heuristic in the family of heuristics.

33. The method of claim 31 further comprising

assigning the reservation slots as standby broadcast slots for use by a same node as the reservation slot assignment using a standby slot heuristic in the family of heuristics.

34. The method of claim 33 further comprising adaptively assigning the broadcast slots and the standby broadcast slots in a broadcast cycle according to network density using an adaptive broadcast cycle heuristic in the family of heuristics.

35. The method of claim 34 further comprising assigning standby broadcast slots on different frequencies when the network density exceeds the capacity of the broadcast cycle using a channelized neighborhood heuristic in the family of heuristics.

36. The method of claim 33 further comprising

allowing neighboring nodes to transmit information in idle standby broadcast slots assigned to a first node using a speculation slot heuristic in the family of heuristics.

37. The method of claim 31 further comprising

segregating neighbors into at least two groups based on the transmission quality of each neighbor using a neighbor segregation heuristic in the family of heuristics.

38. The method of claim 31 further comprising

establishing hard circuits by reserving all slots alone a path from a source node to a destination node before transmitting by using a hard circuit heuristic in the family of heuristics.

39. The method of claim 31 further comprising

establishing soft circuits by assigning enough slots between nodes to satisfying traffic requirements by using a soft circuit heuristic in the family of heuristics.

* * * * *